Signal Processing 125 (2016) 187–202

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

A new RGB image encryption algorithm based on DNA encoding and elliptic curve Diffie–Hellman cryptography

Manish Kumar [a,*], Akhlad Iqbal [c,d], Pranjal Kumar [a,b]

[a] Department of Mathematics, Birla Institute of Technology and Science-Pilani, Hyderabad Campus, Hyderabad 500078, Telangana, India
[b] Department of Electrical Engineering, Birla Institute of Technology and Science-Pilani, Hyderabad Campus, Hyderabad 500078, Telangana, India
[c] Centre for Mathematics and Statistics, School of Basic and Applied Sciences, Central University of Punjab, Bathinda, Punjab 151001, India
[d] Department of Mathematics, Aligarh Muslim University, Aligarh 202002, Uttar Pradesh, India

ARTICLE INFO

Article history:
Received 18 August 2015
Received in revised form
3 January 2016
Accepted 22 January 2016
Available online 6 February 2016

Keywords:
DNA encoding
Elliptic curve
Finite field
Asymmetric key
Image encryption

ABSTRACT

With the increasing use of media in communications, there is a need for image encryption for security against attacks. In this paper, we have proposed a new algorithm for image security using Elliptic Curve Cryptography (ECC) diversified with DNA encoding. The algorithm first encodes the RGB image using DNA encoding followed by asymmetric encryption based on Elliptic Curve Diffie–Hellman Encryption (ECDHE). The proposed algorithm is applied on standard test images for analysis. The analysis is performed on key spaces, key sensitivity, and statistical analysis. The results of the analysis conclude that the proposed algorithm can resist exhaustive attacks and is apt for practical applications.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

In the recent years, due to the increase in digitalization of media, image security has become a prime agenda for secure transmission over unsecured channels. Images are frequently used in diverse areas such as medical imaging, online teaching and advertising, etc. In real-time scenario, transmitting images through internet and storing it on various platforms (cloud server, hard-drive, etc.), the fundamental issue of protecting for confidentiality, integrity, and authenticity is a major concern. Since most of the data is exchanged between anonymous parties. Therefore, there is a need for asymmetric encryption methods.

Elliptic curves for the purpose of cryptography were first proposed individually in [1,2]. ECDHE provides similar level of security with a smaller key size as compared to RSA and DES algorithms. With the recent advanced in DNA technologies, DNA computing has entered in the domain of cryptography (particularly, in the field of image encryption). Various image encryption schemes have been proposed based on DNA encoding such as authors [3–13]. The complimentary rule of DNA has been used to encrypt images [3,6]. Using DNA computation [7] have presented an image encryption algorithm of one-time pad cryptography with DNA strands, which cannot be implemented for frequent use (due to the use of one time pad). A coupled map lattice was used to encrypt images [8], it has been designed by using chaotic map (for simulation purpose, logistic map was used). Recently, Zhang et al. [9] have proposed a new algorithm for image encryption based on multi-chaotic maps using pseudo DNA operation rules to increase complexity and ciphertext unpredictability of the algorithm; [10] have used Lorenz chaotic system along with DNA computing to encrypt color image; [11] have presented a novel image fusion encryption algorithm based on DNA

---

[*] Corresponding author.
E-mail addresses: manish.math.bhu@gmail.com, manishkumar@hyderabad.bits-pilani.ac.in (M. Kumar).

http://dx.doi.org/10.1016/j.sigpro.2016.01.017
0165-1684/© 2016 Elsevier B.V. All rights reserved.

sequence and hyper-chaotic system; and DNA subsequence based couple images encryption algorithm using chaotic system has been proposed in [12]. Likewise, the authors [14] and [15] have proposed a new DNA encoding using chaos maps. Image encryption algorithm using chaotic and hyper-chaotic systems have been researched and reported by several authors, such as [16–20].

However, DNA encoding associated with chaotic (or hyper-chaotic) systems offer good encryption but symmetric in nature and hence unable to perform perfect forward secrecy. In order to overcome the above drawbacks, in this paper, we have proposed a new image encryption algorithm using DNA computing and ECDHE for RGB images which is asymmetric in nature and also provide perfect forward secrecy. The length of shifting sequence offers the users' to chose required lever of security. The proposed algorithm is secure and immune against common attacks (such as known plain-text, chosen cipher-text, cropped and noise attacks). Also, different elliptic curves can be used in the algorithm to vary the need of encryption speed for real-time encryption. Security analysis suggests that the proposed algorithm can be used for secure transmission efficiently.

Table 1
Eight rules of complementary DNA encoding.

Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8
$00-A$	$00-A$	$00-C$	$00-C$	$00-G$	$00-G$	$00-T$	$00-T$
$01-C$	$01-G$	$01-A$	$01-T$	$01-A$	$01-T$	$01-C$	$01-G$
$10-G$	$10-C$	$10-T$	$10-A$	$10-T$	$10-A$	$10-G$	$10-C$
$11-T$	$11-T$	$11-G$	$11-G$	$11-C$	$11-C$	$11-A$	$11-A$

Table 2
Addition and subtraction rule for DNA.

+	A	T	C	G	–	A	T	C	G
G	*G*	*C*	*T*	*A*	*G*	*G*	*T*	*C*	*A*
C	*C*	*A*	*G*	*T*	*C*	*C*	*G*	*A*	*T*
T	*T*	*G*	*A*	*C*	*T*	*T*	*A*	*G*	*C*
A	*A*	*T*	*C*	*G*	*A*	*A*	*C*	*T*	*G*

2. Preliminaries

2.1. DNA encoding

DNA sequencing is performed using four basic nucleic acids namely, Adenine (A), Cytosine (C), Guanine (G), Thymine (T). A and T; C and G are compliments of each other [7]. The total number of possible combinations are $4! = 24$ out of which only 8 follow the complimentary rule as shown in Table 1. In DNA encoding each nucleotide is denoted with a binary number following the complimentary rule, e.g.: $A-00, C-01, G-10, T-11$, so the decimal value 200 (11 00 10 00) will be represented as *TAGA*. The 8-bit pixel values of the image are converted into four 2-bit DNA sequence [21].

2.1.1. DNA sequences: addition and subtraction

Addition and subtraction phenomenon of DNA sequence is very similar to traditional algebraic computations. The addition and subtraction is done over

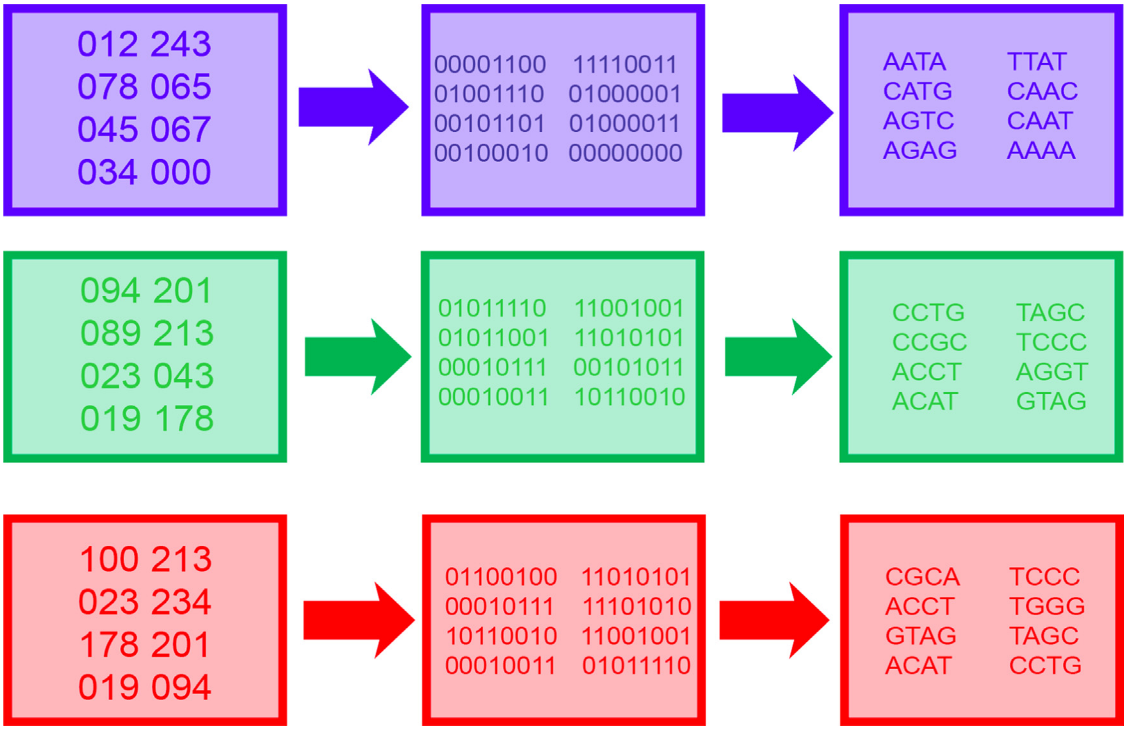

012 243
078 065
045 067
034 000
00001100 11110011
01001110 01000001
00101101 01000011
00100010 00000000
AATA TTAT
CATG CAAC
AGTC CAAT
AGAG AAAA
094 201
089 213
023 043
019 178
01011110 11001001
01011001 11010101
00010111 00101011
00010011 10110010
CCTG TAGC
CCGC TCCC
ACCT AGGT
ACAT GTAG
100 213
023 234
178 201
019 094
01100100 11010101
00010111 11101010
10110010 11001001
00010011 01011110
CGCA TCCC
ACCT TGGG
GTAG TAGC
ACAT CCTG

Fig. 1. DNA encoding of an RGB image.

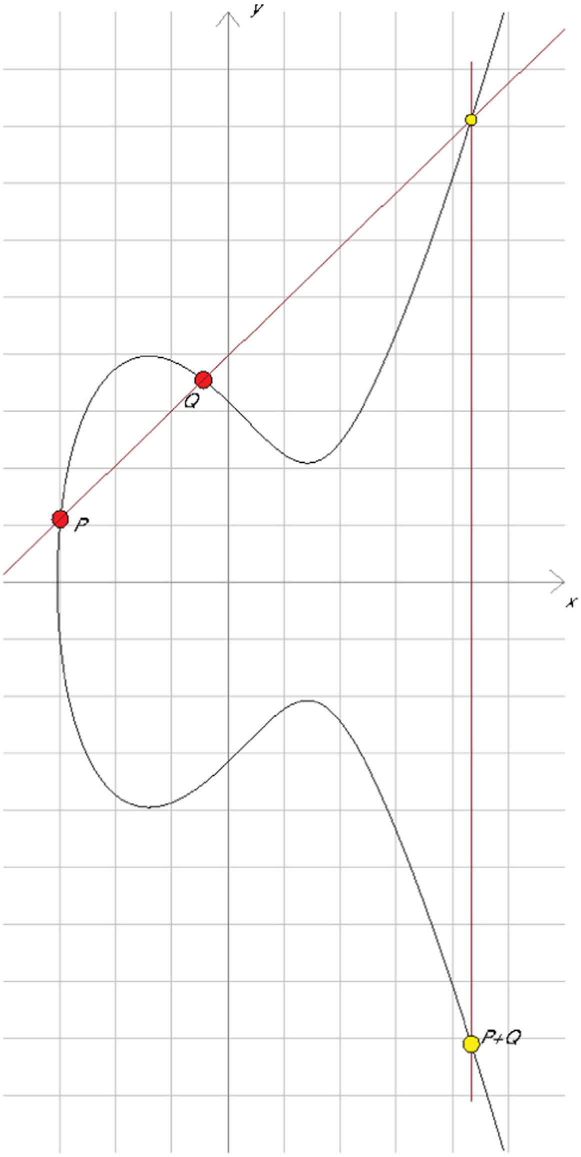


Fig. 2. Elliptic addition of two points on the curve.

modulo 4 [22], adding numbers with DNA, the addition and subtraction are given below in Table 2 when $00-A, 11-T, 01-C$, and $10-G$.

2.2. Elliptic curve cryptography

ECC is a notable mechanism in the field of public key cryptography. ECC makes use of elliptic curves in which the variables and coefficients are elements of a finite field. As compared to the other algorithms (like, RSA and DES) it provides the same level of security with smaller key size. Therefore, it helps in reducing storage and transmission requirements.

2.2.1. Elliptic curve over finite field

From [23], for every prime power p^k there exists a field F_{p^k} with p^k elements. An elliptic curve E is the set of solutions to a generalized Weierstrass equation as follows:

$$E\colon Y^2+a_1XY+a_3Y=X^3+a_2x^2+a_4X+a_6,$$

together with an extra point O. The coefficients a_1, a_2 ,..., a_6 are required to satisfy $\triangle \neq 0$, where the $\triangle$ is defined in terms of certain quantities b_2, b_4, b_6, b_8 given below:

$$b_2=a_1^2+4a_2,\ b_4=2a_4+a_1a_3,\ b_6=a_3^2+4a_6,$$
$$b_8=a_1^2a_6+4a_2a_6-a_1a_3a_4+a_2a_3^2-a_4^2,$$
$$\triangle=-b_2^2b_8-8b_4^3-27b_6^2+9b_2b_4b_6.$$

If $P=(x_1,y_1)$ and $Q=(x_2,y_2)$ are points on E as shown in Fig. 2:

(i) $P=Q=(x_1,-(a_1x_1+a_3)-y_1)$
(ii) If $x_1=x_2$ but $P\neq Q$, then $P\oplus Q=O$
(iii) If $x_1\neq x_2$ and $P\neq Q$, then the line through P and Q is $y=\lambda x+\gamma$, where $\lambda=(y_2-y_1)/(x_2-x_1)$, $\gamma=(y_1x_2-y_2x_1)/(x_2-x_1)$.

Also, if $P=Q$, then the tangent to E at P is $y=\lambda x+\gamma$, where: $\lambda=(3x_1^2+2a_2x_1+a_4-a_1y_1)/(2y_1+a_1x_1-a_3)$, $\gamma=(-x_1^3+a_4x_1+2a_6-a_3y_1)/(2y_1+a_1x_1+a_3)$. Finally, if the line through P and Q (respectively the tangent at $P=Q$) is

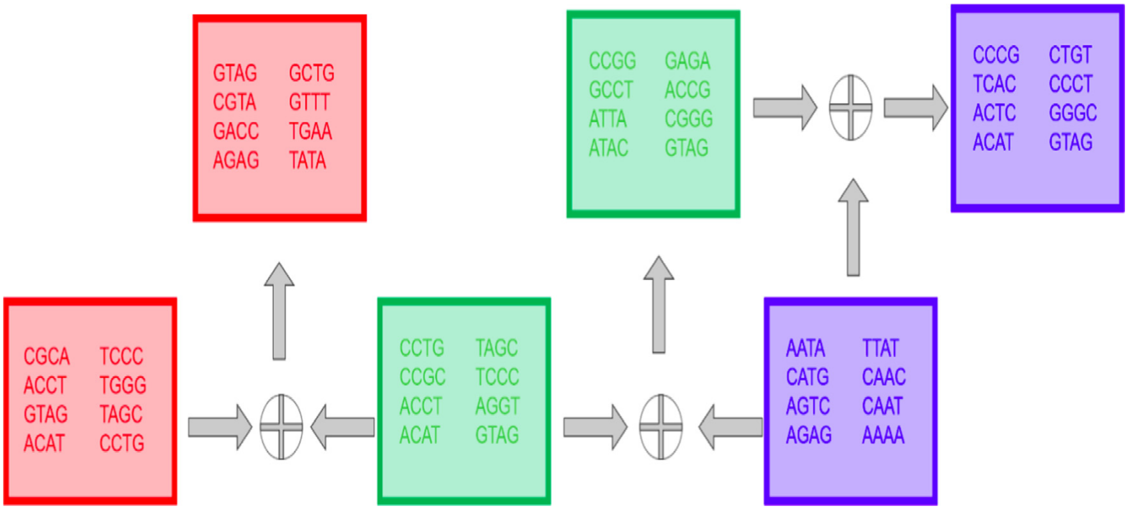

GTAG GCTG
CGTA GTTT
GACC TGAA
AGAG TATA
CCGG GAGA
GCCT ACCG
ATTA CGGG
ATAC GTAG
CCCG CTGT
TCAC CCCT
ACTC GGGC
ACAT GTAG
CGCA TCCC
ACCT TGGG
GTAG TAGC
ACAT CCTG
CCTG TAGC
CCGC TCCC
ACCT AGGT
ACAT GTAG
AATA TTAT
CATG CAAC
AGTC CAAT
AGAG AAAA

Fig. 3. Addition of pixel values using DNA encoding.

$y=\lambda x+\gamma$, then $P\oplus Q=(x_3,y_3)$ where: $x_3=\lambda^2+a_1\lambda-a_2-x_1-x_2$, $y_3=(a_1-\lambda)x_3-\gamma-a_3$.

From Hasse's theorem [23], the number of points on the elliptic curve E over finite field $\mathbb{F}_{p^k}$ denoted by $\#E(\mathbb{F}_{p^k})$ and defined as:

$$p^k+1-2\sqrt{p^k}\le \#E(\mathbb{F}_{p^k})\le p^k+1+2\sqrt{p^k}.$$

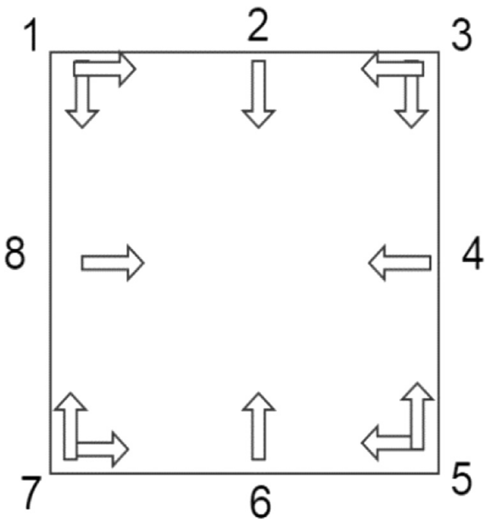


Fig. 4. Process of shifting: 1-Down and right; 2-Down; 3-Down and left; 4-Left; 5-Up and left; 6-Up; 7-Up and right; 8-Left.

2.2.2. *Elliptic curve Diffie–Hellman encryption*

An elliptic curve (a_1,a_2,a_3,a_4,a_6) and a point on the curve (X,Y) belonging to the field $F_2{}^k$ are made publicly available by an authenticating third party. Party A chooses a private value $n_a(0\le n_a\le 2^k-1)$ and computes $Q_a=n_a(X,Y)$ and sends Q_a to Party B. Similarly, Party B chooses a private value $n_b(0\le n_b\le 2^k-1)$ and computes $Q_b=n_b(X,Y)$ and sends Q_b to Party A. Party A then computes $n_aQ_b=n_an_b(X,Y)$ and Party B computes $n_bQ_a=n_an_b(X,Y)$ for which both the values are same and hence act as private keys for encryption.

As we know, the ECDHE offers better security than RSA algorithm. Since ECDHE requires key exchange between parties, it provides perfect forward secrecy. If information needs to be exchanged between multiple parties, each party can provide keys on the same curve and hence remove the need for unique public private key pairs between each parties or a server to moderate the exchange of information as in RSA [24] (for instance, if information needs to be communicated between three parties: Party A, Party B and Party C. Each of the parties will provide their public keys Q_a,Q_b and Q_c and then compute $n_aQ_bQ_c$, $n_bQ_aQ_c$ and $n_cQ_aQ_b$, respectively. Hence the common private key will be $n_an_bn_c(X,Y)$).

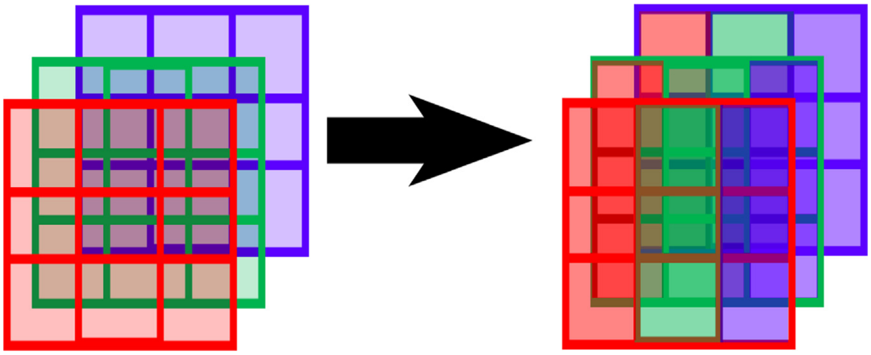


Fig. 5. Interleaving of an RGB image.

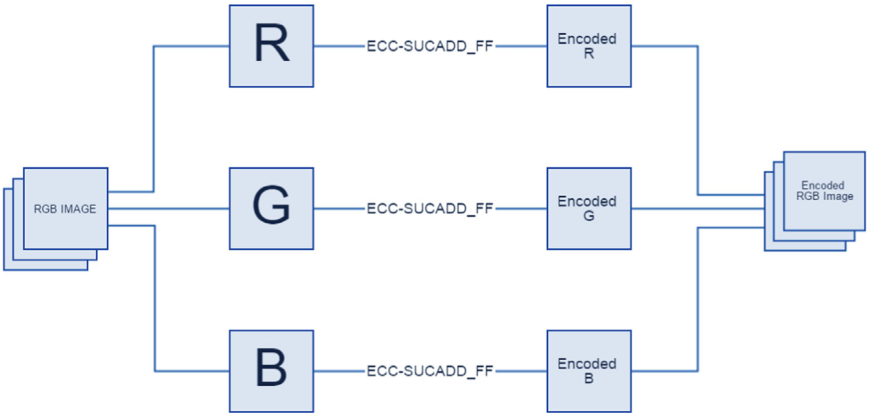


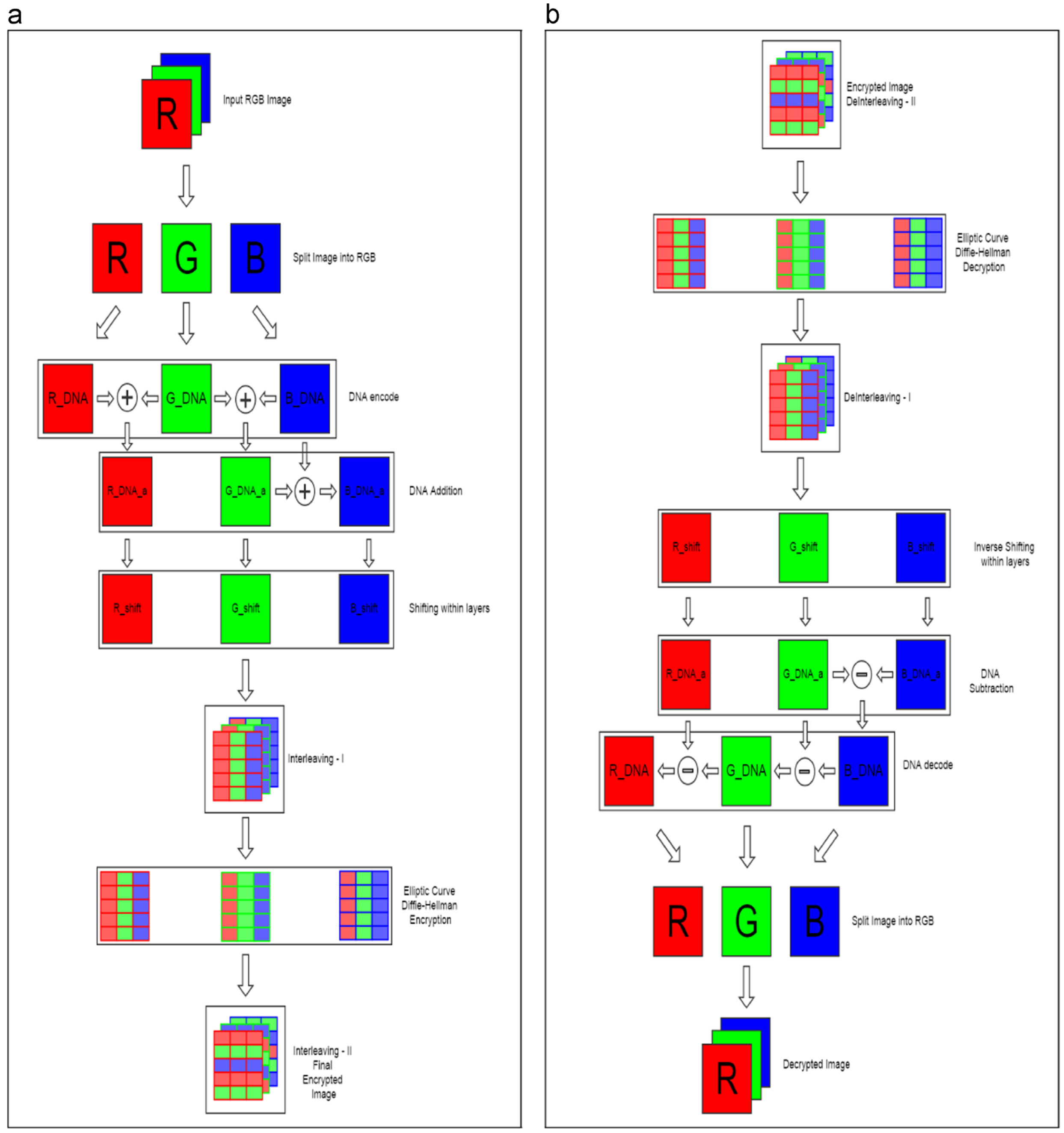
Fig. 6. ECC-SUCADD_FF: Elliptic curve cryptography using successive addition over finite field.

3. Proposed algorithm

An RGB image is decomposed into three layers (Red, Green, Blue) and transformed into DNA sequence matrix. To scramble the image, DNA addition is performed on the layers as shown in Fig. 1. Circular shifting on each layers of the image (can be seen in Fig. 4) is performed to further distort the correlation of the pixel values by scrambling them independently in the spatial domain. This acts as a symmetric key encryption mechanism. The length of shifting sequence can be chosen depending upon the level of security required. The amount of shifting of pixels is obtained by using the shared private keys (say, K_R, K_G and K_B, where the subscripts R, G and B denote each components of an RGB image) from ECC and is fixed for each RGB component on each axis of the image.

The pixels of the image are interleaved to increase the variation in the correlation of the image across the layers in spatial domain [25].

Three different curves and points on these three curves are chosen for each component for an RGB image ECDHE.


a
R
Input RGB Image
R
G
B
Split Image into RGB
R_DNA
G_DNA
B_DNA
DNA encode
R_DNA_a
G_DNA_a
B_DNA_a
DNA Addition
R_shift
G_shift
B_shift
Shifting within layers
Interleaving - I
Elliptic Curve Diffie-Hellman Encryption
Interleaving - II Final Encrypted Image
b
Encrypted Image DeInterleaving - II
Elliptic Curve Diffie-Hellman Decryption
DeInterleaving - I
R_shift
G_shift
B_shift
Inverse Shifting within layers
R_DNA_a
G_DNA_a
B_DNA_a
DNA Subtraction
R_DNA
G_DNA
B_DNA
DNA decode
R
G
B
Split Image into RGB
R
Decrypted Image

Encryption Decryption.

Fig. 7. Flow chart of the proposed algorithm. (a)Encryption. (b) Decryption.

Lookup tables are made available for each curve to speed up the process.

Finally, the RGB image is processed through one more round of interleaving as an added measure to partially hide the implementation of ECDHE.

The process of encryption is stated the steps underneath (can be seen in Fig. 7a).

Step 1: Import the RGB image I $(m \times n \times 3)$, where m and n are the row and column size, respectively.
Step 2: Each component of the RGB is decomposed into binary numbers $(m \times (n \times 8))$ and is encoded using of the DNA encoding scheme to form $(m \times (n \times 4))$ by virtue of selecting one of the eight rules.
Step 3: The DNA addition operation is performed on the components as shown in Fig. 3. $R_{DNA}(i,j)=R(I,j)+G(i,j)$, $G_{DNA}(i,j)=G(i,j)+B(i,j)$ and $B_{DNA}(i,j)=G_{DNA}(i,j)+B(i,j)$.
Step 4: We generate an octal sequence and each layer undergoes circular shift operation using the amount specified by the ECDHE keys and the direction specified in the octal sequence as shown in Fig. 4.
Step 5: The RGB components RRRGGGBBB are interleaved as RGBRGBRGB as shown in Fig. 5.
Step 6: The image is encrypted as $R_{DNA} \times (X_r, Y_r)$ over $E_R(a_{1r}, a_{2r}, a_{3r}, a_{4r}, a_{6r})$, $G_{DNA} \times (X_g, Y_g)$ over $E_G(a_{1g}, a_{2g}, a_{3g}, a_{4g}, a_{6g})$ and $B_{DNA} \times (X_b, Y_b)$ over $E_B(a_{1b}, a_{2b}, a_{3b}, a_{4b}, a_{6b})$ and explained graphically in Fig. 6
Step 7: The RGB components are again interleaved.
Step 8: Finally, RGB components are recombined to get the encrypted image.

The decryption process is similar to the encryption process by taking the reverse order with deinterleaving of the image followed decryption of ECHDE. The image is again deinterleaved and the reordering of the image is done using the same octal sequence but with the negative values of the ECHDE keys. The image finally undergoes DNA subtraction followed the decoding using same rule giving the decrypted image. The decryption algorithm is shown in Fig. 7b.

4. Simulated results

The simulations have been performed on MATLAB environment 8.2 and GNU octave. For instance, a baboon image of size 512 × 512 with the following parameters are used for encryption.

1. R layer: $Y^2+897XY+789Y=X^3+273X^2+321X+672 \in F_{2^{16}}$ $V:(19985, 54699)$ with keys Party $A_{priv}=120$, Party $B_{priv}=130$ and Party $A_{pub}=(25918, 49043)$ and Party $B_{pub}=(58446, 19298)$ giving shared private key $K_R=(13936, 47874)$ (Fig. 8a).
2. G layer: $Y^2+399XY+101Y=X^3+487X^2+9280X+11186 \in F_{2^{16}}$ $V:(33387, 32484)$ with keys Party $A_{priv}=899$, Party $B_{priv}=1798$, and Party $A_{pub}=(15516, 55029)$ and Party $B_{pub}=(32826, 61971)$ giving shared private key $K_G=(65272, 38098)$ (Fig. 8b).
3. B layer: $Y^2+1210XY+72Y=X^3+928X^2+5249X+7892 \in F_{2^{16}}$ $V:(52477, 53228)$ with keys Party $A_{priv}=792$, Party $B_{priv}=685$, and Party $A_{pub}=(400095, 20676)$ and Party $B_{pub}=(57580, 21332)$ giving shared private key $K_B=(48375, 32041)$ (Fig. 8c).

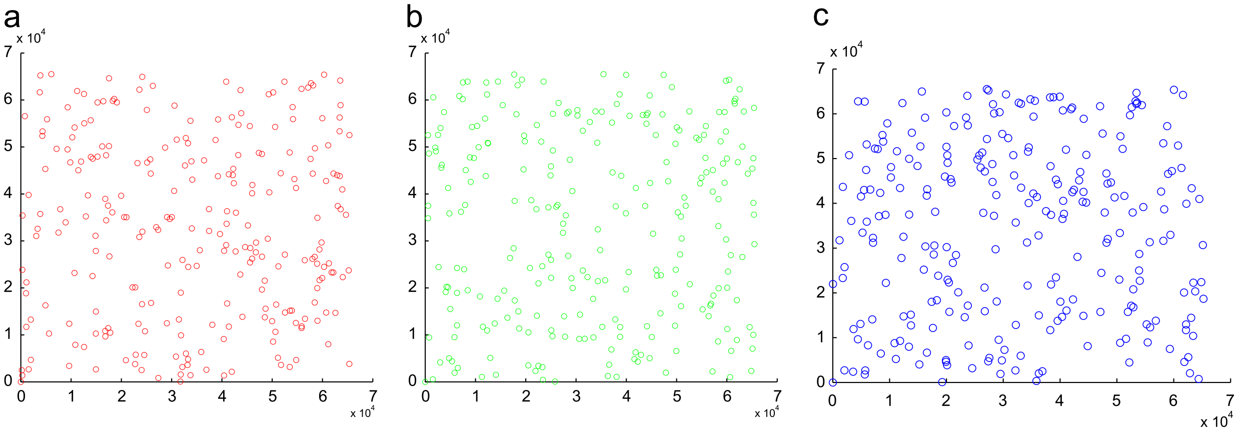


Fig. 8. All shared private keys for elliptic curve Diffie–Hellman encryption: (a) Red component for all pixel values of K_R. (b) Green component for all pixel values of K_G. (c) Blue component for all pixel values of K_B.

Table 3
Octal shifting sequence.

Layer	R	G	B	R	G	B	R	G	B	R	G	B	R	G	B
Position	1	3	5	7	8	6	4	2	3	5	6	4	5	3	4
Dest.	D & R	D & L	U & L	U & R	L	U	R	D	D & L	U & L	U	R	U & L	D & L	U & R

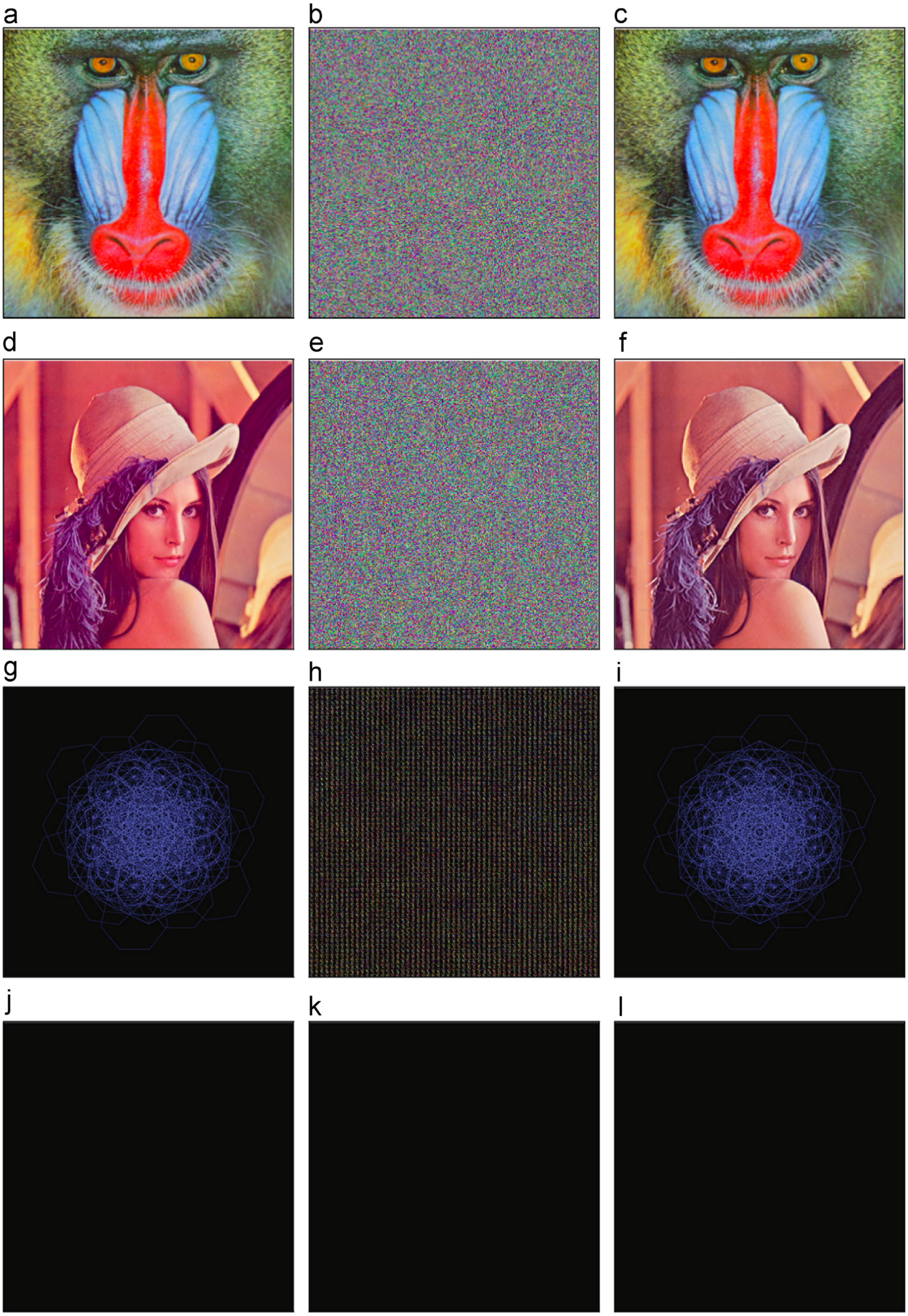


Fig. 9. Simulations. (a) Original Baboon image. (b) Encrypted Baboon image. (c) Decrypted Baboon image. (d) Original Lena image. (e) Encrypted Lena image. (f) Decrypted Lena image. (g) Original hexfractal (sparse) image. (h) Eencrypted hexfractal (sparse) image. (i) Decrypted hexfractal (sparse) image. (j) Original black image. (k) Encrypted black image. (l) Decrypted black image.

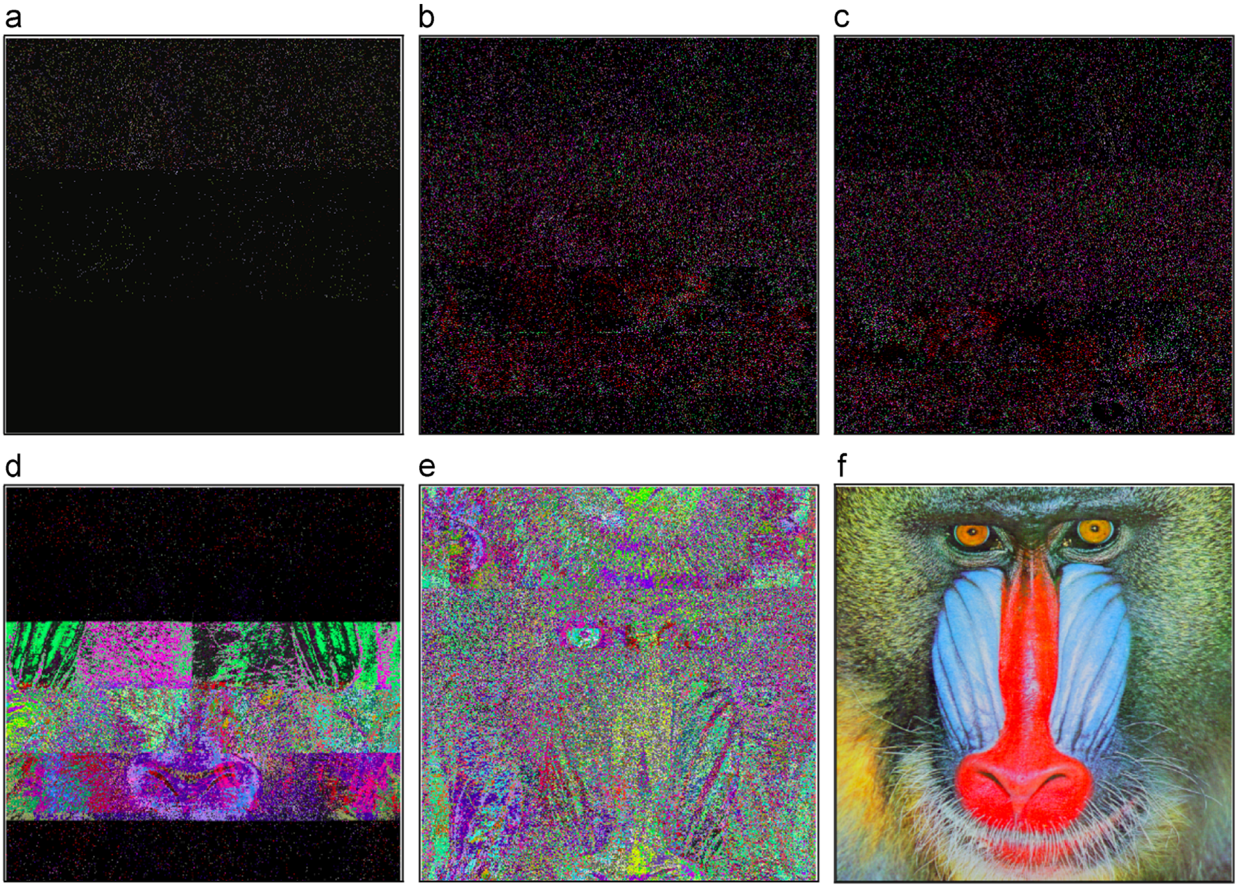


Fig. 10. Key space analysis. (a) Decryption with wrong key sequence and wrong shift sequence. (b) Decryption with wrong keys and correct shift sequence. (c) Decryption with wrong keys and wrong shift sequence. (d) Decryption with wrong key sequence and correct shift sequence. (e) Decryption with correct keys and wrong shift sequence. (f) Correct decryption.

The octal shifting sequence can be seen in Table 3.

Fig. 9 shows that the proposed algorithm is good for encrypting color images. Since the proposed algorithm is asymmetric in nature, and with the inclusion of extra layer of security such as DNA computing and second interleaving increases the robustness of the proposed algorithm. Moreover, DNA computing along with a second interleaving hide ECDHE making it harder to compromise. For fast encryption, curve25519 proposed by Bernstein [26] can be used.

5. Security analysis

5.1. Key space analysis

To sustain security of images, the key space should be large enough to resist an exhaustive attacks. The key space for the proposed algorithm is as follows:

(Number of DNA complement rule) × (Number of points on Elliptic curve)^6 × (Number of possible shifting operations)^n

$$8 \times \left(2^k+1-2\sqrt{2^k}\right)^6 \times 8^n \le (\mathit{Key\ Space}) \le 8 \times \left(2^k+1+2\sqrt{2^k}\right)^6 \times 8^n.$$

For the purpose of simulation, we have taken the field $\mathbb{F}_{2^{16}}$ and the length of shifting sequence is fifteen ($n = 15$). Hence, the key space lies between $2^{143.9322}$ to $2^{144.0674}$, which provides high security and robustness against brute force. The key space can be enhanced with respect to the field size (k) and the length of the shifting sequence (n).

5.2. Key sensitivity analysis

For a viable image encryption algorithm, the decryption should fail with the slightest change in the key should not allow correct decryption. Various permutations in the key space are applied and the results are shown in Fig. 10.

5.3. Statistical attacks

5.3.1. Histogram analysis of encrypted image

Histograms provide a clear illustration of how the pixels intensity is distributed. Figs. 11 and 12 show the histograms of the original (Fig. 9a and d), encrypted (Fig. 9b and e) and decrypted (Fig. 9c and f) images. The histogram of the encrypted image is rigorously diverse from the original images' histogram. Further the encrypted images' histogram look similar and hence cannot be used for decryption.

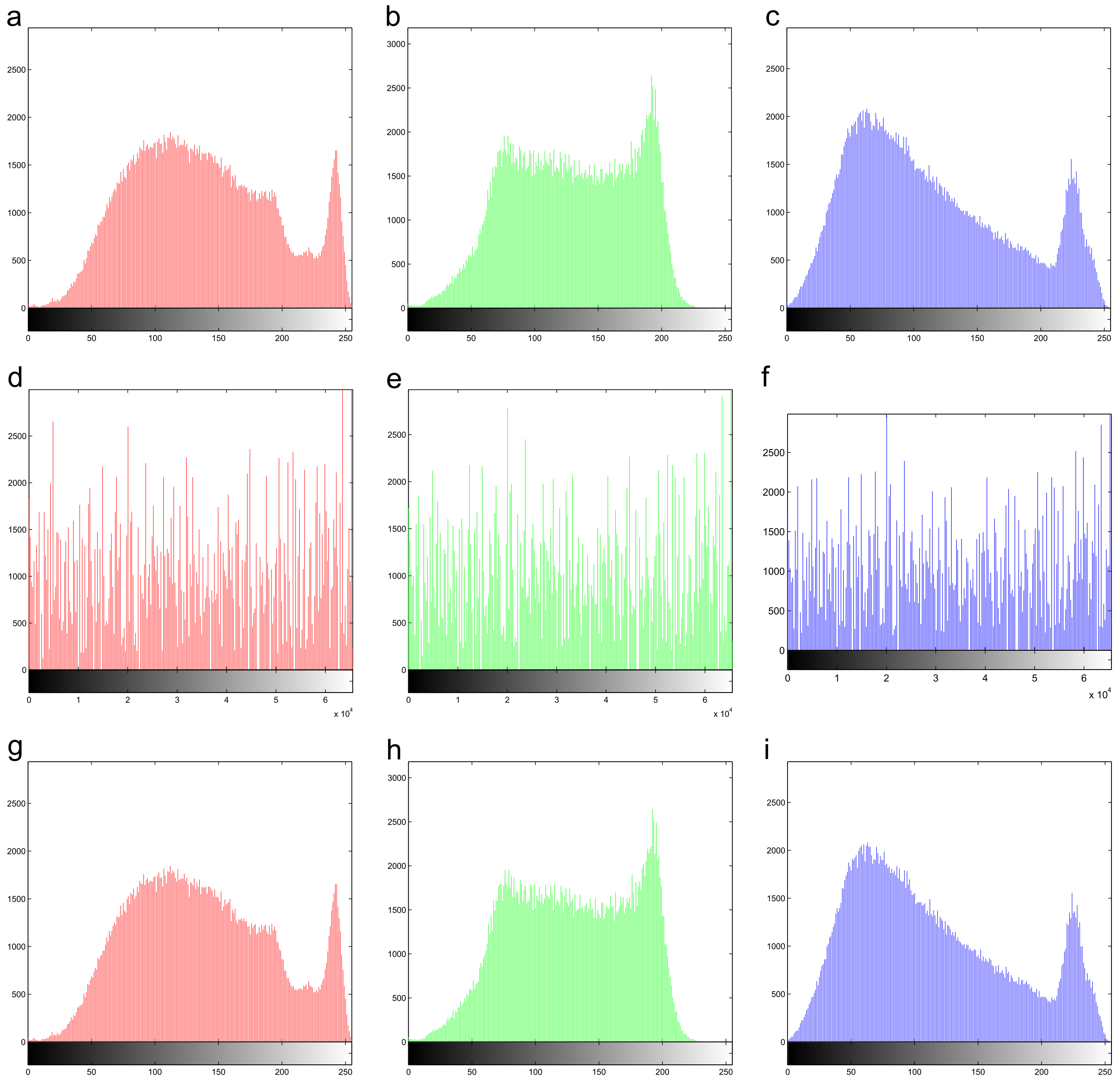


Fig. 11. Histograms analysis. (a) R component of the original image Fig. 9a. (b) G component of the original image Fig. 9a. (c) B component of the original image Fig. 9a. (d) R component of the encrypted image Fig. 9b. (e) G component of the encrypted image Fig. 9b. (f) B component of the encrypted image Fig. 9b. (g) R component of the decrypted image Fig. 9c. (h) G component of the decrypted image Fig. 9c. (i) B component of the decrypted image Fig. 9c.

5.4. Correlation of two adjacent pixels

Correlation between two adjacent pixels can be tested in four ways, by taking two vertically adjacent pixels or by taking two horizontally adjacent pixels or by taking two diagonal adjacent pixels or taking two anti-diagonal adjacent pixels in the encrypted image. We have selected 5000 random pairs of adjacent pixels for testing purpose. The results are shown in Tables 4–11. The correlation factor is calculated for the original Baboon image (Fig. 9a); original Lena image (Fig. 9d) and the encrypted Baboon image (Fig. 9b); encrypted Lena image (Fig. 9e) as

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D_x}\sqrt{D_y}};\ cov(x,y) = E[(x - E(x))(y - E(y))];$$

$$E(x) = \frac{1}{L}\sum_{i=1}^{L} x_i;\ D_x = \frac{1}{L}\sum_{i=1}^{L}(x_i - E(x))^2,$$

where x and y are the value of the adjacent pixels and L is the number of samples taken. The scatter plot of the horizontal correlation is shown in Figs. 13 and 14. Further, from the comparison Table 15 can be inferred that the proposed algorithm offers good security.

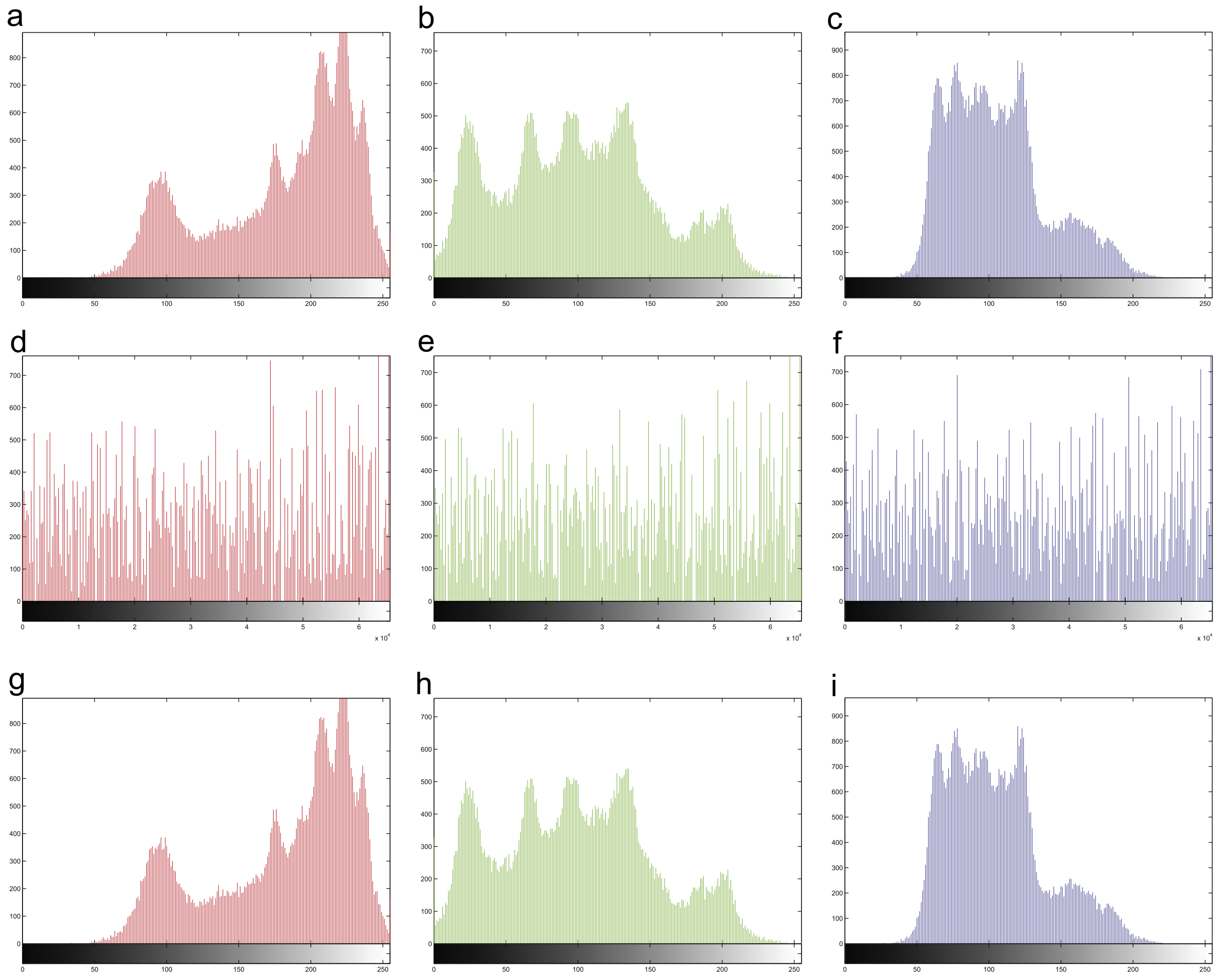


Fig. 12. Histograms analysis. (a) R component of the original image Fig. 9d. (b) G component of the original image Fig. 9d. (c) B component of the original image Fig. 9d. (d) R component of the encrypted image Fig. 9e. (e) G component of the encrypted image Fig. 9e. (f) B component of the encrypted image Fig. 9e. (g) R component of the decrypted image Fig. 9f. (h) G component of the decrypted image Fig. 9f. (i) B component of the decrypted image Fig. 9f.

Table 4
Correlation coefficients of two adjacent pixels in the plane-image and the corresponding cipher-image of Baboon image in horizontal direction.

Component of the image	Original image	Encrypted image
Red component of the RGB image	0.928080547623810	0.018695496141379
Green component of the RGB image	0.862574287003796	0.006687141505165
Blue component of the RGB image	0.908792871958637	0.006791010171259

5.5. *Mean square error*

The Mean Square Error (MSE) between the original image and encrypted image is calculated by the formula given below:

$$MSE = \frac{1}{N \times M} \sum_{n=1}^{N} \sum_{m=1}^{M} \left[|f(i,j) - f_0(i,j)|^2 \right],$$

where f and f_0 are the intensity functions of decrypted and original images and (i,j) represents the position of the pixels. Since the original and the decrypted image are the same the MSE of decrypted image with respect to the original image is zero.

5.6. *Peak signal to noise ratio*

The Peak Signal Noise Ratio (PSNR) of a given color component is the ratio of the mean square difference of the component for the two images to the maximum mean square difference that can exist between any two images. Greater the PSNR value, better the image quality. For encrypted image, smaller value of PSNR is expected. PSNR values have been calculated for the red, green and blue components of the cipher-images with respect to their plain-images (Tables 12–13). The low PSNR values reflect the difficulty in retrieving the plain-images from the

Table 5
Correlation coefficients of two adjacent pixels in the plane-image and the corresponding cipher-image of Baboon image in vertical direction.

Component of the image	Original image	Encrypted image
Red component of the RGB image	0.865075313937753	−0.006449278290027
Green component of the RGB image	0.769719865770650	0.016421558279165
Blue component of the RGB image	0.885904418624641	0.001296864065258

Table 6
Correlation coefficients of two adjacent pixels in the plane-image and the corresponding cipher-image of Baboon image in diagonal direction.

Component of the image	Original image	Encrypted image
Red component of the RGB image	0.853801965078605	−0.001305310128727
Green component of the RGB image	0.725686678327742	0.009244980804621
Blue component of the RGB image	0.842708189500707	0.017197010308042

Table 7
Correlation coefficients of two adjacent pixels in the plane-image and the corresponding cipher-image of Baboon image in anti-diagonal direction.

Component of the image	Original image	Encrypted image
Red component of the RGB image	0.853354625495345	0.005877851836138
Green component of the RGB image	0.718744447679737	−0.020538983270572
Blue component of the RGB image	0.839496270088290	−0.008756002392316

Table 8
Correlation coefficients of two adjacent pixels in the plane-image and the corresponding cipher-image of Lena image in horizontal direction.

Component of the image	Original image	Encrypted image
Red component of the RGB image	0.932667029586456	0.003535161409385
Green component of the RGB image	0.922281493233349	−0.009706871635486
Blue component of the RGB image	0.893825219391821	0.018571551282381

cipher-image, without the knowledge of secret key:

$$PSNR = 20 \times \log\frac{255^2}{\sqrt{MSE}}.$$

5.7. Robustness against known-plaintext attack and chosen-ciphertext attack

Image encryption algorithm should be robust against all types of cryptanalytic. In this subsection, we discuss the security analysis of the proposed algorithm as known-plaintext attack and chosen-ciphertext attack. In known-plaintext attack a cryptanalyst has an access to a plaintext and the corresponding ciphertext and try to derive a correlation between these two or by applying the same key try to decrypt ciphertext for the encrypted plaintext. In chosen-ciphertext attack a cryptanalyst tried to find a matching plaintext for the ciphertext chosen randomly. This proposed algorithm is immune against the known-plaintext attack, and chosen-ciphertext attack for the reason that decryption process depends not only on the combination of the possible correct keys but also on the correct permutation of the key sequence. If attacker knows about all the possible exact keys, but unaware about correct order of the key sequence, attacker cannot decipher the image correctly.

Table 9
Correlation coefficients of two adjacent pixels in the plane-image and the corresponding cipher-image of Lena image in vertical direction.

Component of the image	Original image	Encrypted image
Red component of the RGB image	0.962408763131362	−0.004076324725952
Green component of the RGB image	0.954693507829754	0.005312032028056
Blue component of the RGB image	0.934300708191352	0.010691355134514

Table 10
Correlation coefficients of two adjacent pixels in the plane-image and the corresponding cipher-image of Lena image in diagonal direction.

Component of the image	Original image	Encrypted image
Red component of the RGB image	0.907966009398585	−0.041028497842243
Green component of the RGB image	0.880408776021311	−0.008522993738987
Blue component of the RGB image	0.863453990135567	−0.017458909511577

Table 11
Correlation coefficients of two adjacent pixels in the plane-image and the corresponding cipher-image of Lena image in anti-diagonal direction.

Component of the image	Original image	Encrypted image
Red component of the RGB image	0.936121841285154	−0.001725957229897
Green component of the RGB image	0.914472528539722	0.010691070462096
Blue component of the RGB image	0.879969883231512	0.001156870814213

5.8. Cropped attack analysis

The cropped attack analysis has been performed to check the robustness against data loss for real-time applications of the proposed algorithm [16,27,28]. Two different types of cropped images have been decrypted

Table 12
MSE and PSNR of the encrypted image with respect to the original image of Baboon image.

Component of the image	MSE	PSNR
Red component of the RGB image	4.902866021067581e+08	−38.739700949687872
Green component of the RGB image	4.894815488766505e+08	−38.732563949902605
Blue component of the RGB image	4.908507314666850e+08	−38.744695118938409

Table 13
MSE and PSNR of the encrypted image with respect to the original image of Lena image.

Component of the image	MSE	PSNR
Red component of the RGB image	1.481197079758423e+09	−43.575324862807250
Green component of the RGB image	1.480929875633255e+09	−43.574541336110620
Blue component of the RGB image	1.469906628458771e+09	−43.542093874632386

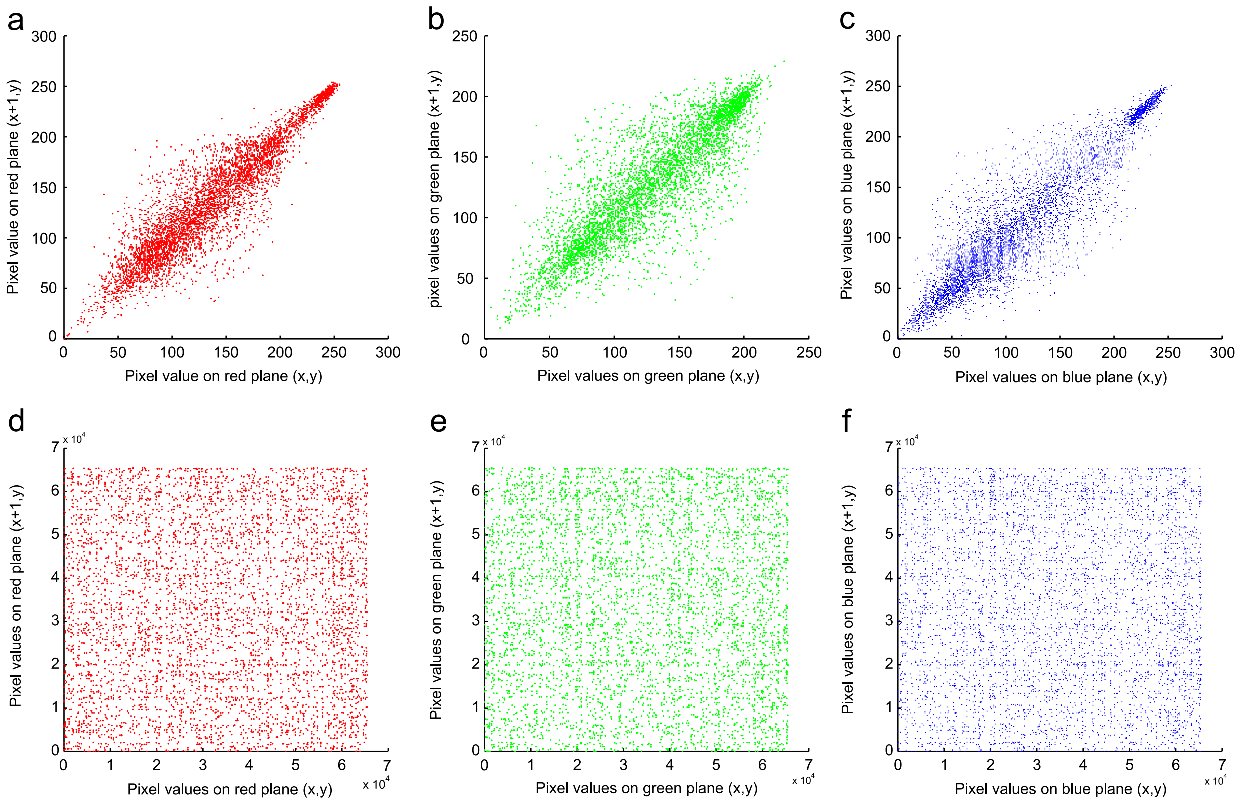


Fig. 13. Horizontal correlation of the baboon image. (a), (b) and (c) are the RGB plots of the original image. (d), (e) and (f) are the RGB plots of the encrypted image.

using the correct set of keys. One image contains cropping on different layers (R, G, and B) in the image, whereas the second image contains cropping on sections where there is total loss of data. The main objective of the analysis is to make sure that the decrypted image can be recognized visually. From Fig. 15, we can see that the proposed algorithm is robust against cropped attack.

5.9. Transmission noise analysis

Transmission loss analysis has been performed to analyze the robustness of the algorithm to the noise that might be picked up during real-time transmission of images. The analysis is performed by introducing *salt and pepper* noise in the encrypted image synthetically. The image is then decrypted using the correct set of keys and the decrypted image is checked for visual recognition. From Fig. 16, it can be established that the proposed algorithm is invulnerable against noise.

5.10. Computational complexity analysis

To evaluate the computational complexity of the proposed algorithm, the time consuming part is the DNA

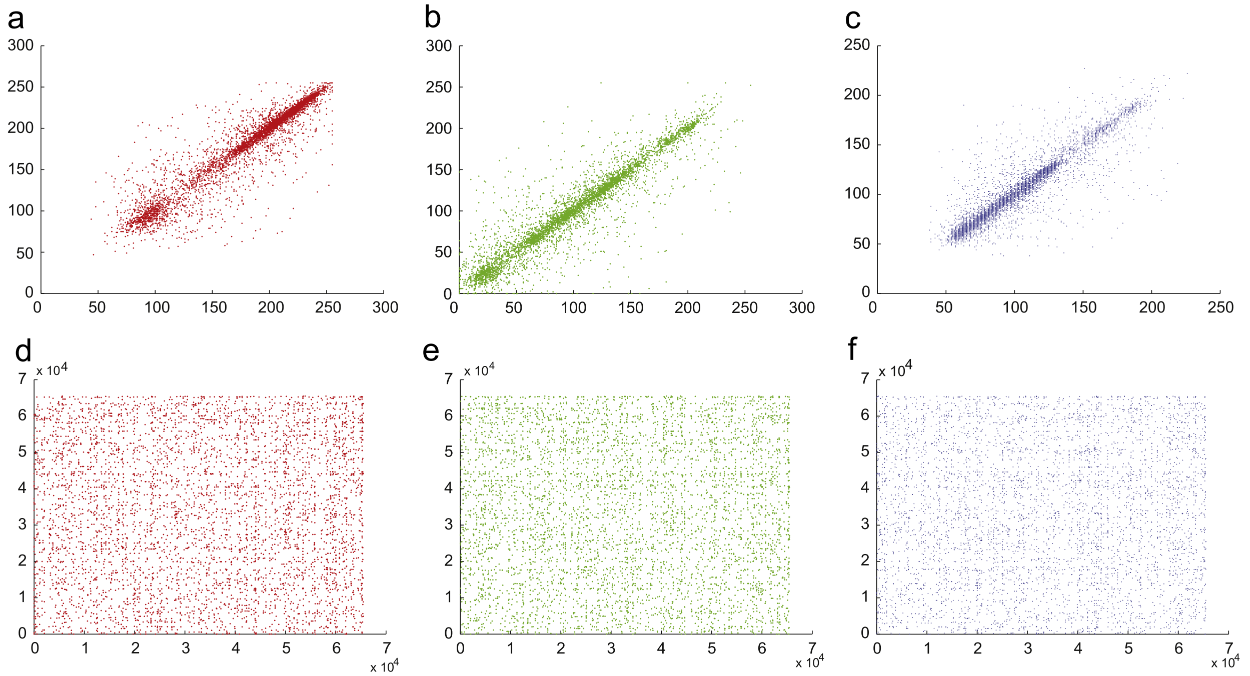


Fig. 14. Horizontal correlation of the Lena image. (a), (b) and (c) are the RGB plots of the original image. (d), (e) and (f) are the RGB plots of the encrypted image.

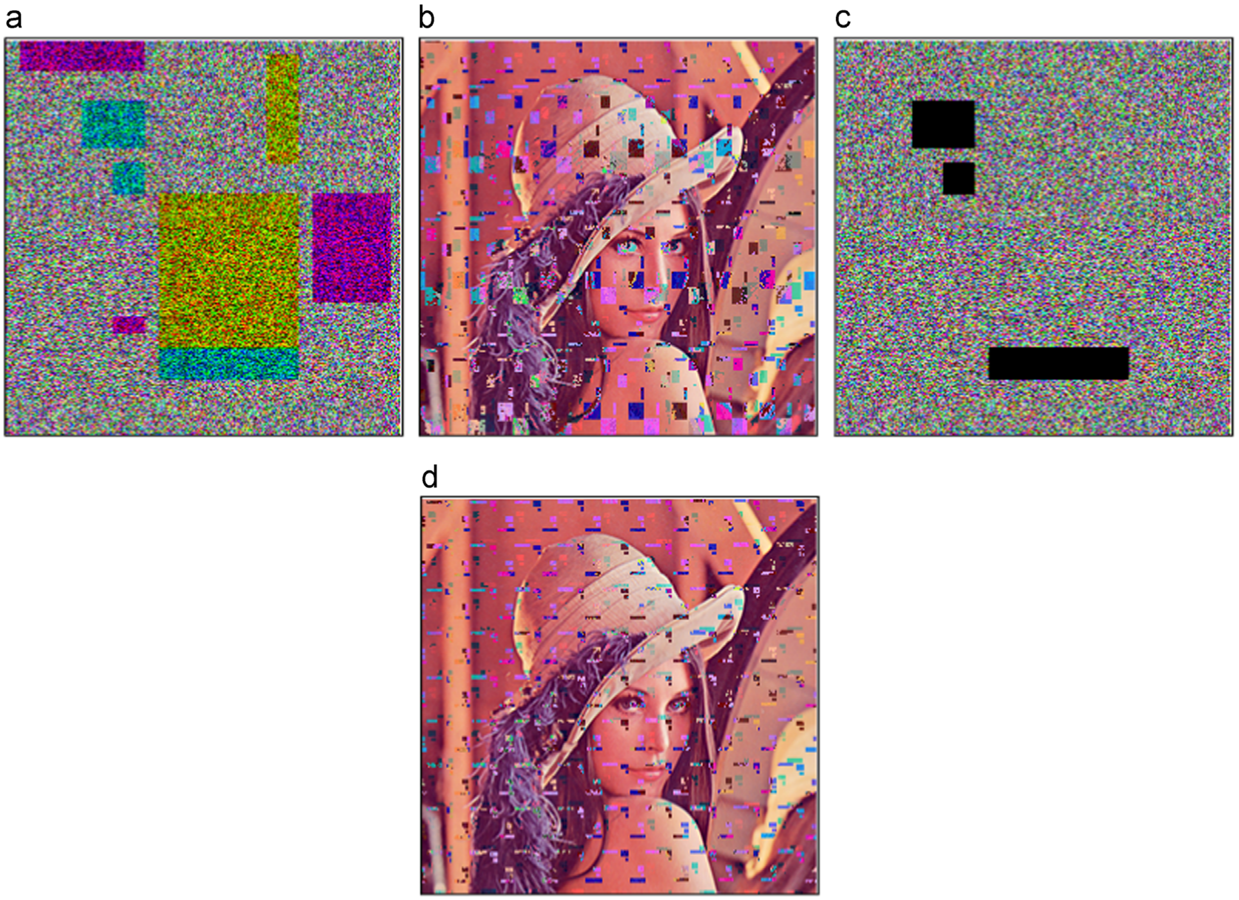


Fig. 15. Cropped attack experiment. (a) Encrypted image with different locations from each layer (R, G, and B) is cropped. (b) Decrypted result. (c) Encrypted image with same locations from each layer (R, G, and B) is cropped. (d) Decrypted result.

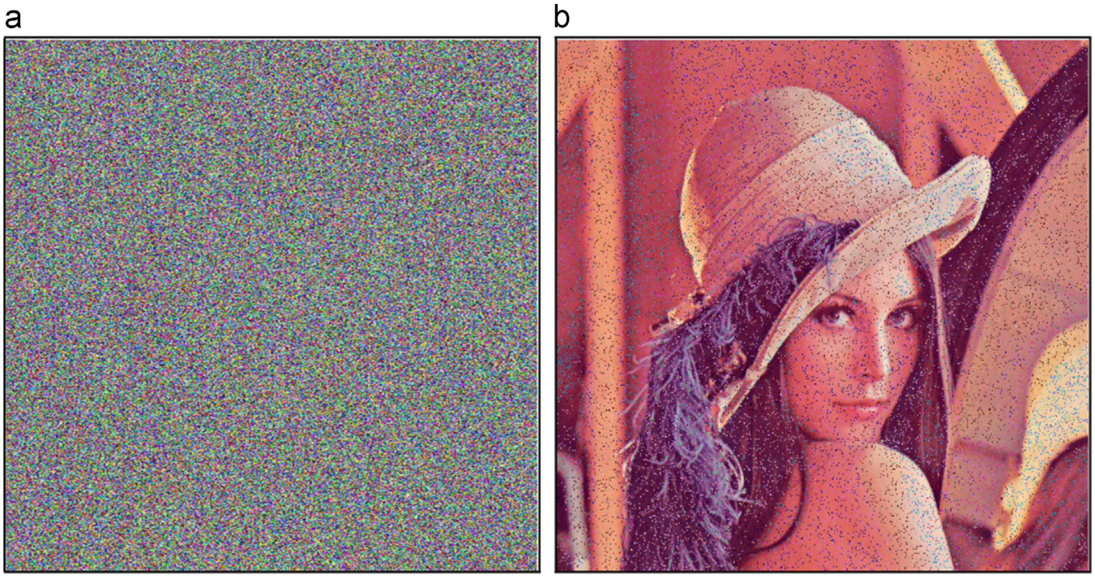


Fig. 16. Transmission noise attack analysis. (a) Noise attack image. (b) Decrypted result.

Table 14
Comparison of authors [7,14,15,25,29–32] with our approach.

S.no.	authors [7,14,15,25,29–32]	Our approach
1.	[29,15] can only encrypt greyscale images	The proposed algorithm encrypt RGB image
2.	[30,7] require DNA based biological experiments	This algorithm is free from biological experiments
3.	[14,25] have used symmetric encryption	The proposed approach is asymmetric
4.	The algorithm [31] increases size of the original image	Original and encrypted images are same size
5.	Large key has been used [32]	Small keys are used which provides huge key space

encoding operations as a DNA computer has been simulated for the proposed algorithm. Each pixel is decomposed into DNA nucleotides in $\bigcirc(M \times N \times 3)$ where M and N are the dimensions of a RGB image. Then the pixel nucleotides now $M \times N \times 3 \times 4$ are added in $\bigcirc(M \times N \times 3 \times 4)$. Finally the image is reconstructed in complexity of $\bigcirc(M \times N \times 3)$. The shifting and shuffling operations are both done in $\bigcirc((M+X) \times (N+Y) \times 3)$ as they are linear in nature giving a complexity value of $\bigcirc(M \times N \times 3)$. Since a lookup table is created for ECDH encryption, the complexity is $\bigcirc(M \times N \times 3)$. Finally, the computational complexity of the proposed algorithm is $\bigcirc(M \times N \times 3)$ and is linear in nature and the encryption time depends on the size of the image. The running time of the algorithm can be reduced by performing the encryption on a DNA based computer. As well as implementing *single process multiple data* scheme as majority of the DNA encoding and ECDH are independent operations.

6. Comparison

The proposed approach has been compared with the existing techniques in Table 14. Further, security analysis suggests that the proposed algorithm is efficient, robust and can encrypt RGB images as compared to [15,29] which can only encrypt greyscale images. Table 15 shows that the proposed technique has superior correlation between pixels, hence provides robust security against statistical attacks [10,17,25,32,33]. As we can infer from Section 5.8 (cropped attack analysis) and Section 5.9 (transmission noise analysis) that, the proposed algorithm can resist loss in data during transmission and can be implemented for real time transmission as compared to [8–18,20], in these algorithms, chaotic maps have been implemented and any loss of data can limit or prohibit visual recognition of the decrypted image extensively. The proposed algorithm also retains the original size of the image after encryption as compared to [31] which increases the size after encryption. Unlike [14,25] which are symmetric, the proposed algorithm is asymmetric in nature and can be implemented for exchange of sensitive images between anonymous parties. The MSE and PSNR obtained in the proposed algorithm (shown in Tables 16 and 17) are better than results of [25,33]. Therefore, the proposed algorithm is robust than the compared algorithms.

7. Conclusion

In this paper, we have proposed a new and secure RGB image encryption algorithm using DNA computing and ECDHE. An image is encoded using the DNA compliment rule and is scrambled by shifting and interleaving operations, thereafter encrypted using ECDHE. The image finally undergoes another round of interleaving giving the final encrypted image. The use of DNA encoding and ECDHE provide a dual layer of security. The key space of the algorithm is large and can be varied according to need. The decryption is successful only when the keys are applied in a particular order hence the knowledge of just the keys are insufficient for decryption. The statistical analysis shows

Table 15
Comparison of horizontal correlation of encrypted image of Lena image with other authors [10,17,32,33].

Component of the image	Proposed algorithm	Zhang et al. [10]	Wang et al. [17]	Murillo et. al [32]	Mishra et al. [33]	Kumar et al. [25]
Red component of the RGB image	0.003535	−0.0065	−0.010899	0.0135	0.0219	0.0181
Green component of the RGB image	−0.009706	0.0009	−0.018110	−0.0835	−0.0046	−0.0067
Blue component of the RGB image	0.018571	−0.0008	−0.006104	−0.0170	−0.0211	0.0154

Table 16
Comparison of MSE value of decrypted image using the proposed algorithm with [25,33].

Component of the image	Proposed algorithm	Kumar et al. [25]	Mishra et al. [33]
Red component of the RGB image	0	1.281718	$8.3014X10^{-26}$
Green component of the RGB image	0	0.945639	$1.0696X10^{-25}$
Blue component of the RGB image	0	1.233145	$3.1629X10^{-25}$

Table 17
Comparison of PSNR value of decrypted image using the proposed algorithm with [25,33].

Component of the image	Proposed algorithm	Kumar et al. [25]	Mishra et al. [33]
Red component of the RGB image	∞	47.052878	298.939290
Green component of the RGB image	∞	48.373549	297.838589
Blue component of the RGB image	∞	47.220662	303.129948

that the algorithm provides good security and can protect against common attacks. Therefore, this proposed algorithm can be used for secure image transmission.

Acknowledgments

All the authors are deeply grateful to the Editor-in-Chief Dr. Björn Ottersten and Handling Editor Dr. Nam Ik Cho for smooth and fast handling of the manuscript. The authors would also like to thank the anonymous referees for their valuable, genuine comments and suggestions to improve the quality of this work. One of the authors [PK] would like to thank for the support of the research grant provided by Department of Science and Technology, New Delhi, Government of India under INSPIRE grant no. 861/2011.

References

[1] N. Koblitz, Elliptic curve cryptosystems, Math Comput 48 (177) (1987) 203–209.
[2] V. Miller, Use of elliptic curves in cryptography, Lect Notes Comput Sci 85 (1) (1985) 417–426.
[3] T. Head, G. Rozenberg, R.S. Bladergroen, C.K.D. Breek, P.H. M. Lommerse, H.P. Spaink, Computing with DNA by operating on plasmids, Biosystems 57 (2) (2000) 87–93.
[4] Y. Niu, X.Y. Wang, An anonymous key agreement protocol based on chaotic maps, Commun Nonlinear Sci Numer Simulat 16 (4) (2011) 1986–1992.
[5] K. Halvorsen, W.P. Wong, Binary DNA nanostructures for data encryption, PLOS ONE 7 (9) (2012) e44212.
[6] X.D. Zheng, J. Xu, W. Li Parallel, DNA arithmetic operation based on *n*-moduli set, Appl Math Comput 212 (1) (2009) 17–184.
[7] A. Gehani, T.H. LaBean, J.H. Reif, DNA-based cryptography, DIMACS series in discrete mathematics, Theoret Comput Sci 54 (1) (2000) 23–249.
[8] X.Y. Wang, Y.Q. Zhang, X.M. Bao, A novel chaotic image encryption scheme using DNA sequence operations, Optics Lasers Eng 73 (1) (2015) 53–61.
[9] Q. Zhang, L. Liu, X. Wei, Improved algorithm for image encryption based on DNA encoding and multi-chaotic maps, Int J Electron Commun 68 (3) (2014) 186–192.
[10] Q. Zhang, X. Wei, RGB color image encryption method based on Lorenz chaotic system and DNA computation, IETE Tech Rev 30 (5) (2013) 404–409.
[11] Q. Zhang, L. Guo, X. Wei, A novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system, Optik 124 (2013) 3596–3600.
[12] Q. Zhang, X. Wei, A novel couple images encryption algorithm based on DNA subsequence operation and chaotic system, Optik 124 (23) (2013) 6276–6281.
[13] P. Zhen, G. Zhao, L. Min, X. Jin, Chaos-based image encryption scheme combining DNA coding and entropy, Multimed Tools Appl http://dx.doi.org/10.1007/s11042-015-2573-x.
[14] L. Liu, Q. Zhang, X. Wei, A RGB image encryption algorithm based on DNA encoding and chaos map, Comput Electr Eng 38 (5) (2012) 1240–1248.
[15] H. Liu, X. Wang, A. Kadir, Image encryption using DNA complementary rule and chaotic maps, Appl Soft Comput 12 (5) (2012) 1457–1466.
[16] X.Y. Wang, L. Yang, R. Liu, A. Kadir, A chaotic image encryption algorithm based on perceptron model, Nonlinear Dyn 62 (3) (2010) 615–621.
[17] X.Y. Wang, L. Teng, X. Qin, A novel color image encryption algorithm based on chaos, Signal Process 92 (4) (2012) 1101–1108.
[18] H.J. Liu, X.Y. Wang, Color image encryption based on one-time keys and robust chaotic maps, Comput Math Appl 59 (10) (2010) 3320–3327.
[19] Y.Q. Zhang, X.Y. Wang, A symmetric image encryption algorithm based on mixed linear-nonlinear coupled map lattice, Inform Sci 273 (10) (2014) 329–351.
[20] H.J. Liu, X.Y. Wang, Color image encryption using spatial bit-level permutation and high-dimension chaotic system, Optics Commun 284 (16–17) (2011) 3895–3903.
[21] H.J. Shiu, K.L. Ng, J.F. Fang, R.C.T. Lee, C.H. Huang, Data hiding methods based upon DNA sequences, Inform Sci 180 (11) (2010) 2196–2208.
[22] P. Wasiewicz, J.J. Mulawka, W.R. Rudnicki, B. Lesyng. Adding numbers with DNA, IEEE International Conference on Systems Man and Cybernetics 1 (2000) 265–270.
[23] J. Hoffstein, J. Pipher, J.H. Silverman, An Introduction to Mathematical Cryptography, Springer Science + Business Media, New York, 2008, 279–339.
[24] R. Rivest, A. Shamir, L. Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Commun ACM 21 (2) (1978) 120–126.

[25] M. Kumar, P. Powduri, A. Reddy, An RGB image encryption using diffusion process associated with chaotic map, J Inf Secur Appl 21 (1) (2015) 20–30.

[26] D.J. Bernstein, New Diffie–Hellman Speed Records Public Key Cryptography (PKC), 3958, Springer, Berlin, Heidelberg, 2006, 207–228.

[27] M. Kumar, D.C. Mishra, R.K. Sharma, A first approach on an RGB image encryption, Optic Lasers Eng 52 (2014) 27–34.

[28] Y. Li, F. Zhang, Y. Li, R. Tao, Asymmetric multiple-image encryption based on the cascaded fractional Fourier transform, Optics Lasers Eng 72 (2015) 18–25.

[29] N. Kang, A Pseudo DNA Cryptography Method arXiv:0903.2693 [cs. CR] (2009).

[30] C.T. Celland, V. Risca, C. Bancroft, Hiding messages in DNA microdots, Nature 399 (1999) 533–534.

[31] J.B. Lima, L.F.G. Novaes, Image encryption based on the fractional Fourier transform over finite fields, Signal Process 94 (2014) 521–530.

[32] M.A. Murillo-Escobar, C. Cruz-Hernanandez, F. Abundiz-Perez, P. M. Lopez-Gutierrez, O.R. Acosta Delcampo, A RGB image encryption algorithm based on total plain image characteristics and chaos, Signal Process 109 (2015) 119–131.

[33] D.C. Mishra, R.K. Sharma, Manish. Kumar, Kuldeep. Kumar, Security of color image data designed by public key cryptosystem associated with 2D-DWT, Fractals 22 (4) . 1450011-1 1450011-16.